Computational Methods for Quality Check, Preprocessing and Normalization of RNA-Seq Data for Systems Biology and Analysis

Mazzoni, Gianluca; Kadarmideen, Haja N.

Published in:
Systems Biology in Animal Production and Health

DOI:
10.1007/978-3-319-43332-5_3

Publication date:
2016

Computational Methods for Quality Check, Preprocessing and Normalization of RNA-Seq Data for Systems Biology and Analysis

Gianluca Mazzoni and Haja N. Kadarmideen

Abstract

The use of RNA sequencing (RNA-Seq) technologies is increasing mainly due to the development of new next-generation sequencing machines that have reduced the costs and the time needed for data generation.

Nevertheless, microarrays are still the more common choice and one of the reasons is the complexity of the RNA-Seq data analysis. Furthermore, numerous biases can arise from RNA-Seq technology, and these biases have to be identified and removed properly in order to obtain accurate results.

Nowadays, many tools have been developed which allow to perform each step without high-level programming skills. However, each step of the pipeline needs to be performed carefully and requires a good knowledge of both the technology and the algorithms.

In this comprehensive review, we describe the fundamental steps of the pipeline for RNA-Seq analysis to identify differentially expressed genes: raw data quality control, trimming and filtering procedures, alignment, postmapping quality control, counting, normalization and differential expression test.

For each step, we present the most common tools and we give a complete description of their main characteristics and advantages focusing on the statistics that they perform and the assumptions that they make about the data.

The choice of the right tool can have a big impact on the final results. Until now, no gold standard has been established for this type of analysis.

G. Mazzoni (✉) • H.N. Kadarmideen
Department of Large Animal Sciences, University of Copenhagen,
Groennegaardsvej 7, Frederiksberg C 1870, Denmark
e-mail: gianluca.mazzoni@sund.ku.dk

H.N. Kadarmideen (ed.), *Systems Biology in Animal Production and Health, Vol. 2*,
DOI 10.1007/978-3-319-43332-5_3

In conclusion, this review can be useful for both educational purposes as well as for less experienced practitioners of animal genomic research. In the absence of a commonly accepted standard procedure, the general overview presented in this review can help to make the best choices during the implementation of an RNA-Seq pipeline.

1 Introduction

Next-generation sequencing (NGS) technologies allow the generation of huge quantities of biological data. The development of new NGS machines has led to a reduction in costs and the time needed for data generation. In transcriptomics, the use of RNA sequencing technologies is ever increasing. RNA-Seq has considerably more benefits than microarray technology: it does not rely on previous knowledge and annotation, it has a wide range of sensitivity in detecting transcripts and it allows to quantify expression of different isoforms, study specific allele expression and identify new transcripts (Zhao et al. 2014).

The advantages on RNA sequencing compared to microarray technologies are even more valuable in systems genetics and system biologies studies.

RNA sequencing data facilitates delving into the analysis and extracting information about biological pathways and gene function.

Nevertheless, microarrays are still the more common choice for gene expression profiling and for differentially expressed genes analysis. The reasons are many.

The cost is still significantly higher for RNA-Seq than microarrays. Furthermore, RNA-Seq data brings with it logistic challenges, for example, the high storage capacity needed for the huge quantity of raw data produced as well as the computational power needed to perform some steps of the bioinformatics pipeline (Zhao et al. 2014).

Furthermore, RNA-Seq data is more complex, and a good knowledge of the technology and its related aspects are necessary in order to produce reliable results.

Different biases and artifacts that arise from these technologies and specific statistics have to be applied to obtain consistent and reliable results.

Nowadays, there are many tools available to perform all the different steps of the bioinformatics pipeline of RNA-Seq data (Garber et al. 2011). Some of them have a graphical interface which allows researchers with a basic computational background to perform all the steps to the final results. However, a good knowledge of the algorithm and a computational background is still necessary to obtain accurate results and make the correct choices in term of tools and statistical tests. Tools differ in the statistics that they perform and in the assumptions that they make about the data. Therefore, they can be more or less efficient with regard to specific characteristics of the dataset as well as the experimental design.

The basic steps of the bioinformatics pipeline for RNA-Seq data are: raw data quality control followed by trimming and filtering procedures, alignment, postmapping quality control, counting and normalization statistic test for differential expression (Mutz et al. 2013) (Fig. 1).

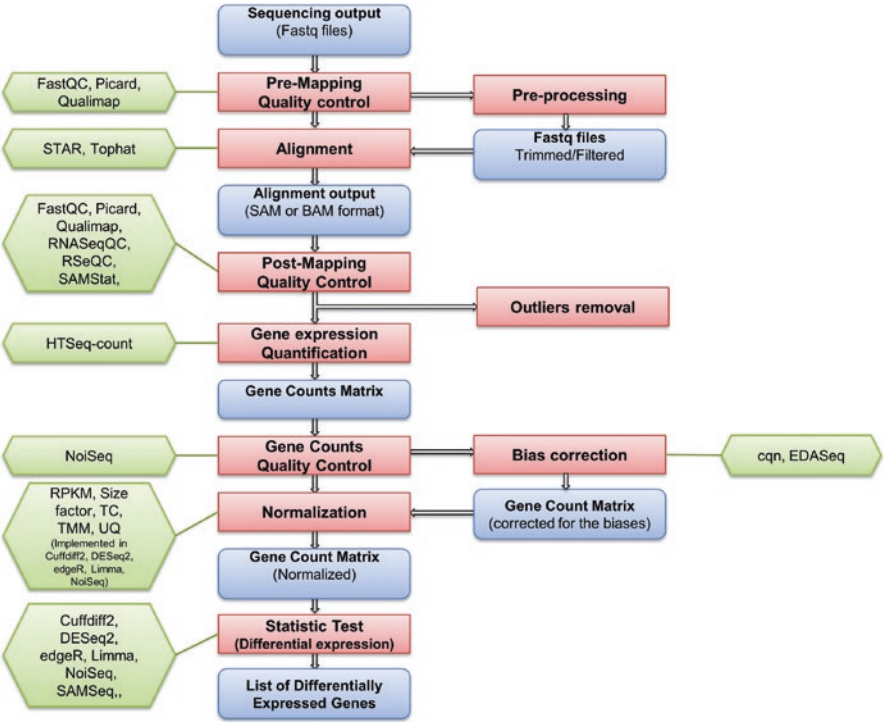

Fig. 1 This picture represents the basic RNA-Seq data analysis pipeline. The *red boxes* are the main steps. The *blue boxes* describe the type of file that is given as input or produced as output at each step. The *green boxes* contain the list of the tools described in the text and they are connected to the step that they perform

2 Raw Data Quality Control

Raw data from RNA-Seq technology is a text file with a FASTQ format. The biological sequences of the reads as well as the sequencing quality values at each nucleotide base are stored in this file. Sequencing quality changes along the positions of the reads usually with a machine specific trend (Fig. 2a).

This bias together with contaminations of unwanted reads and PCR artifacts, GC content and presence of adapters represent technical biases.

Quality control of the raw data is a very important step that facilitates the detection of biases generated during the sequencing procedure that, if not correctly removed, can generate problems like incorrect mapping during the alignment and affect the final results.

The more common tools used in this step are FastQC (Andrews 2010), Qualimap (García-Alcalde et al. 2012) and Picard Tools (Wysoker et al. 2012). These tools are easy to use and the first two also have graphical interfaces for users with no computational skills. The statistics that are usually considered at this step are: total number

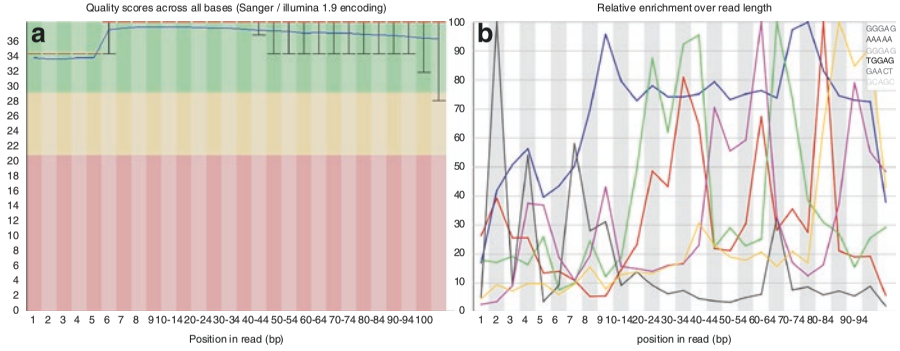

Fig. 2 (**a**) Per base sequence quality plot obtained with FastQC. The RNA sample obtained from bovine cumulus cells has been sequenced with Illumina technology. The reads with low average quality have been filtered out, but it is still possible to see the typical trend where the quality tends to decrease moving along the read length. (**b**) Kmer content computed with FastQC. The *plots* represent the top overrepresented kmers in the sample, across the read positions

of reads, per base sequence quality, per sequence quality score, per base sequence content, per sequence GC content, per base N content, sequence duplication levels, overrepresented sequences and kmer content. This type of quality control is the same as that applied to DNA sequencing data. It is not RNA-specific and it can only provide information about the quality of read data related to NGS technologies.

Bases with low sequencing quality have a higher probability to be wrong. Regions where the quality is too low could have many mistakes that occurred during the sequencing and should be trimmed or filtered out. On the other hand, (Williams et al. 2016) recently found that a too aggressive trimming of RNA-Seq data before gene expression quantification can have great impact on the final estimation leading to unpredictable changes, mainly caused by the generation of very short reads.

Tools like Picard, FastQC or Qualimap compute the summary statistics at each position considering a representative subset of the reads. They generate a boxplot for each position of the read to represent the distribution of the quality per position.

Once identified, this type of issue can be removed by trimming specific regions or entire reads, considering different criteria chosen on the basis of the quality trend of the library.

GC content distribution and overrepresented sequence statistics point out the presence of contaminations or PCR artifacts, or problems during the library preparation.

If the library preparation is carried out correctly, it is expected to have a specific distribution of GC across the set of reads. If the distribution is different from the expected one, it is because there is an overrepresentation probably due to contaminations.

With regard to the level of contamination, if most of the library is represented by contaminations, the sample should be removed, but first it would be better to test whether it is an outlier by using clustering techniques or exploratory analysis such as principal component analysis (PCA). Otherwise, if the contamination represents only a small portion of the library and the sample does not turn out to be an outlier, the contamination can be identified and removed before proceeding with the analysis.

The kmer content is another way to identify biases due to the sequencing or the library preparation technology. The graph represents the overrepresentation of specific sub-sequences along the length of the reads. Library protocols based on random priming have a specific imbalance at the start of the library (Fig. 2b).

Overrepresented reads in the library can be due to strongly expressed transcripts, contaminations, PCR artifacts, adapter content or DNA sequences used during the lab work. Furthermore, they can represent rRNA transcripts that have not been correctly depleted during the RNA purification step. To identify the origin of the overrepresented reads, the sequences can be aligned against RNA sequences in publicly available databases using BLAST or compared against UniVec, an annotated database for vector sequences provided by NCBI (Cochrane and Galperin 2010).

3 Alignment

During this step, the read sequences of the cDNA fragments originating from the random fragmentations and retrotranscription of RNA transcripts are aligned to reference genomes (Wang et al. 2009).

In this way, it is possible to identify the gene or the genomic locus that gave origin to the transcript from which each fragment derived.

The choice of the aligner has to be made considering the library and sequencing protocol as well as the objective of the analysis.

Tophat (Kim and Salzberg 2011) and STAR (Dobin et al. 2013) are two aligners specific for RNA-Seq data (able to identify splicing sites), which have shown the best performances.

Tophat and STAR have been tested together with other aligners using different datasets and they showed similar accuracy (Engström et al. 2013), but the latter has the advantage of being much faster and in the case of large datasets, it can be the best solution.

In the case of de novo mapping, the reads are used to generate contigs and reconstruct the set of isoforms for a specific gene present in a sample directly from the sequenced reads. The process can be performed by using a reference or based only on the reads (Garber et al. 2011).

The set of contigs obtained can then be used as a reference to count the reads that map on them and quantify their expression in the sample. A well-known tool for de novo mapping is Trinity.

Trinity is composed of three independent software modules: Inchworm, Chrysalis and Butterfly. As a final output, the tool gives a full-length transcript with the corresponding alternatively spliced isoforms (Grabherr et al. 2011).

4 Postmapping Quality Control

Postmapping quality control is a fundamental step that allows to identify issues that have occurred during the sequencing or sample extraction or library preparation that can be identified only after alignment.

Nowadays, there are many freely available tools that are able to perform postmapping quality control.

These tools do not have a direct impact on the final results; however, it is fundamental to check the samples before proceeding with the other steps of the pipeline (Williams et al. 2014). Some of the tools are very user-friendly and furthermore, they generate easily interpretable outputs compared to others that need more computational skills.

The most widely used tools are FastQC, Picard Tools, Qualimap, RNA-SeqQC (DeLuca et al. 2012), RSeQC (Wang et al. 2012) and SAMStat (Lassmann et al. 2011).

During the postmapping quality control, two main types of statistics can be performed: general statistics similar to the one applied to raw data and RNA-Seq specific statistics. The first type focuses on NGS-related problems (number of reads

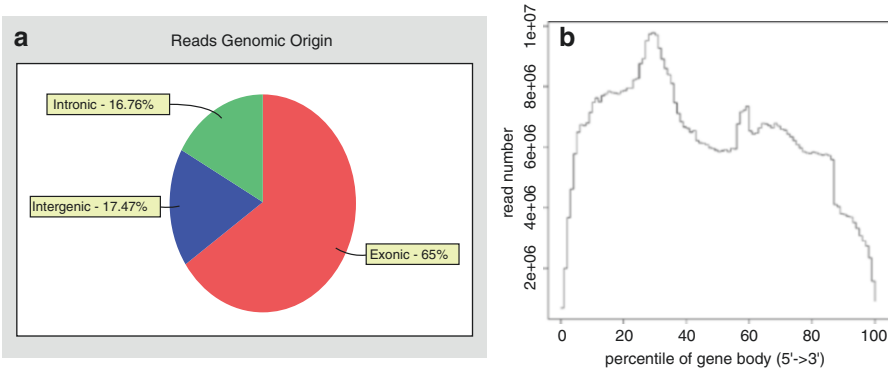

Fig. 3 (**a**) Pie chart obtained with Qualimap showing the percentages of reads mapped to exonic, intronic and intergenic regions of RNA-Seq data from bovine samples. The computation is based on a General Feature Format file where all the information about genomic features of the species of interest were annotated; in this case, we used *Bos taurus* UMD v.3.1.83. (**b**) Gene body coverage computed with RSeQC. The plot represents the coverage along the length of all the transcripts annotated in the bovine genome, normalized from 1 to 100. The reduction present at the 3´ of the transcript indicates a low level of degradation present in the sample

mapped, nucleotide composition, GC percentage, kmer bias) with the only difference being that the statistics are based only on uniquely mapped reads.

FastQC and Picard Tools can also be used at this point of the analysis together with other tools like SAMStat.

SAMStat performs a deeper analysis to detect possible biases related to the mapping quality.

This tool generates a plot where the properties of unmapped, poorly mapped and accurately mapped reads are compared in order to see if some differences are related to the quality of the alignment.

RNA-Seq postmapping statistics focus on genome coverage, intron/exon coverage, intron/exon junction analysis, and in the case of paired end protocols the insert size distribution (Fig. 3a).

Considering that our reads are generated mainly from processed transcripts, especially in the case of mRNA-enriched libraries, we expect that most of them will map to previously annotated exonic regions related to intronic and even less intergenic regions.

These types of statistics are organism-specific because they are strictly dependent on the level of annotation of the genome and obviously on the library protocol used.

Unexpected percentages of reads from intronic and intergenic regions point out problems during library preparation or contamination.

Another important analysis is the intron/exon junction percentages (known, partially known, novel junction). If the sequencing is deep enough and is a good representation of the sample, the spliced junctions should be rediscovered in an RNA-Seq experiment. Spliced junction saturation analysis is also implemented in RSeQC.

The introns/exons junction saturation is computed by re-sampling and thus increasing the total number of reads; thereby computing each time the percentage of known junctions identified.

This information is dependent on the annotation of the genome, but it is important to understand whether the information contained in the data is enough to perform differential splicing.

In the case of paired end protocols, the insert size distribution can be useful to check if the alignment ran correctly.

This statistic has to be specific for RNA, because to compute the correct insert size distribution, the presence of introns when the paired reads are mapped back to the genome have to be considered.

If the sequenced fragment has originated from two exons and the splicing site is in the middle between the forward and the reverse reads, the real insert size can be obtained by subtracting the length of the intron from the distance between the reads' mapping site in the genome.

Insert size statistics are implemented in Picard Tool, Qualimap, RNA-SeqQC and RSeQC. While the first three extract it directly from the SAM file, RSeQC performs a more complex computation, taking the possible presence of introns between two paired reads into consideration.

RSeQC and Qualimap are able to compute an interesting postmapping quality control called gene body coverage. This test is useful, especially in cases where samples have problems in the quality and integrity of the RNA.

The tools give as output a graph representing the level coverage across the length of the transcripts present in the genomes, normalized from 1 to 100 (Fig. 3b).

Qualimap, together with Picard Tools, provides a module specific for RNA sequencing and together with RSeQC and RNA-SeqQC represent the most complete tools for postalignment quality control in RNA-Seq data.

RNA-SeqQCs can also perform a multisample comparison providing information such as correlations and GC content comparisons among samples.

Some tools are less intuitive, while other packages like Qualimap have a well developed graphical interface and provide a complete, well-organized graphical output particularly useful for researchers with weak computational skills.

The ideal way to get a complete impression of the data is to combine the results from different tools, exploiting the advantages of each of them.

This concept is implemented in a recently developed tool called Quality Control for RNA-Seq (QuaCRS) (Kroll et al. 2014). The tool runs FastQC, RNA-SeqQC and SeQC and merges results in an easily interpretable and accessible way.

5 Counting

In this step, reads that map under a biological feature of interest are counted in order to quantify its expression in a sample. Various tools perform this step. The differences are few among these types of tools and they are related mainly in the different ways of considering reads that overlap more than one feature. The estimation of the

expression can be made at different levels for different biological features (gene level, transcript level, exonic level), or it can be applied to all the transcripts identified during de novo mapping.

For example, HTSeq (Anders et al. 2014) and Cufflinks (Trapnell et al. 2013) are commonly used tools to perform this step.

6 Normalization

Even if RNA sequencing was initially considered completely immune of biases, normalization is still a fundamental step (Wang et al. 2009).

It facilitates the removal of biases and it is necessary in order to obtain accurate results during the comparison both within and between samples.

Normalization is tricky and complex in RNA-Seq data, as there are different bias types to take into consideration. In RNA-Seq experiments, biases can be of two types: within-sample bias that is due mainly to gene length bias and GC content bias, and between-sample biases due to the sequencing depth (Dillies et al. 2013).

The gene length bias originates because longer transcripts likely generate a higher number of fragments and consequently a higher number of reads. Thus, it is likely to have a higher level of expression rather than shorter transcripts due to this technical problem and not due to a real activation or inactivation of the transcription (Zheng et al. 2011; Oshlack and Wakefield 2009).

Similar problems occur in fragments with different GC contents (Risso et al. 2011).

GC-rich and GC-poor fragments result in being underrepresented in RNA sequencing, which leads to biases at the gene expression level (Benjamini and Speed 2012).

To make things even more complicated, it has been seen that GC content bias is not consistent between samples. It is lane-dependent and probably introduced during the library preparation step (Risso et al. 2011).

Until now, it has not been determined which method performs better in normalizing for GC content bias.

One of the methods used to account for length bias is the RPKM unit (reads per kilobase of exon per million fragments), which divides the discrete counts of the reads by the total number of reads sequenced and by the length of the transcript and then computes the proportion to one million total reads (Mortazavi et al. 2008). In this way, the expression value of a gene is independent on the length of its transcripts.

Various tools are able to correct for this type of bias, like EDASeq (Risso et al. 2011) and cqn (Hansen et al. 2012) where the GC bias or length bias are included as covariates.

The correction for the biases is dependent on the objective of the study.

If the objective is to rank genes within a sample, for example, to identify which genes are more active in a specific cell type, the biases that must be checked are gene length and GC content.

On the other hand, if the experiment is designed to compare gene expressions between samples to identify differentially expressed genes, the most influential bias to consider is the difference in the library size.

The library size, computed as the total number of reads in a sample, can lead to false positives or false negatives during the analysis, as more reads will be assigned to each gene if a sample is sequenced to a greater depth.

However, it is also very important to consider that gene length and GC content also have an effect in between-sample comparisons; in fact, genes with higher counts are more likely to be defined as differentially expressed than genes with lower counts.

Cqn and EDASeq have been developed in such a way that they correct first for within-sample effect of the GC content and then they correct for between-sample bias.

It has been seen that normalization for library size with simple scaling is not enough. Together with sequencing depth and gene length, the composition of the RNA population has to be considered.

If the majority of genes are highly expressed in one condition compared to the other, the results of the analysis will be skewed (Robinson and Oshlack 2010).

More sophisticated normalization methods have been developed to correct for differences in library size (Oshlack et al. 2010).

Normalization methods have been tested with different datasets (Dillies et al. 2013).

The method implemented in the DESeq2 package (Anders and Huber 2010), together with trimmed mean of M values (TMM) (Robinson and Oshlack 2010) showed good precision and sensibility in false positive rates and power of detection. The first methods use scaling factor for each sample, computed as the median of the ratio between genes and their respective geometric mean computed across samples, while TMM removes the genes that are most expressed and with the highest log ratios and using the remaining genes, a scaling factor is computed as the weighted mean of log ratios between the sample and a reference.

Other methods are also used with good performances, such as upper quartile (Bullard et al. 2010), where gene counts are divided by the upper quartile of the gene counts and median where gene counts are divided by the median of the gene counts.

Even if RPKM, as explained earlier, takes into account the gene length, this method together with total count (TC), in which the counts of the genes are divided by the total number of reads in the sample, is indicated to be ineffective.

The performance of a normalization method is strictly dependent on the dataset. In some cases, no differences have been found in the final results between various methods (Seyednasrollah et al. 2015).

In general, there is no agreement on which is the best method and it is very important to check if the normalization applied worked fine on a dataset. This can be achieved by comparing the median and the distribution of gene expression across genes. In this way, it is possible to identify batch effect on the samples. We expect that after normalization, if the procedure is performed correctly, the distributions should have similar medians and distributions across samples. A similar test is

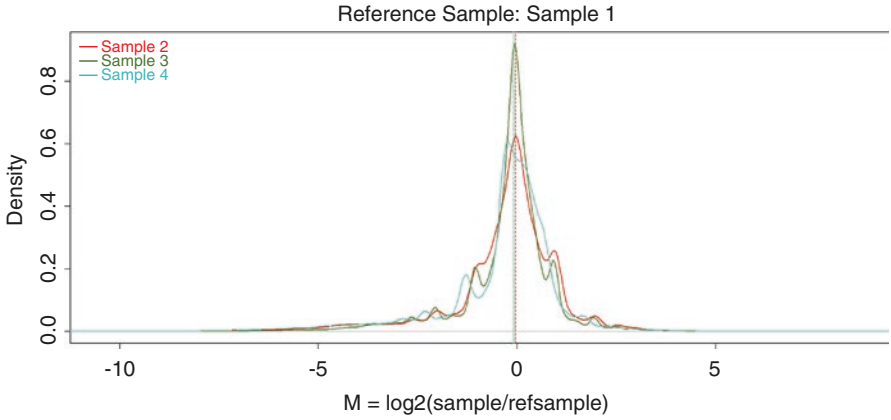

Fig. 4 NOISeq batch effect exploration graph. A sample is used as a reference and NOISeq compares the distributions and the medians among all the samples. The RNA samples analyzed are obtained from bovine cumulus cells, sequenced with Illumina technology using the same library preparation. The samples need to be normalized before proceeding with the next step of the analysis. This issue is mainly due to differences in library sizes

provided by NOISeq (Tarazona et al. 2012). This R package compares read distributions among samples using a sample as a reference, check for presence of GC content bias and length bias (Fig. 4).

Once the data are correctly normalized and transformed, various exploratory analyses can be performed and systems genetic approaches can be applied.

Normalization has less influence in the case of co-expression analysis because we focus on the correlations between expression levels of pairs of genes across all the samples.

In any case, tools for co-expression analysis suggest normalizing the data and applying logarithm-based transformations. For example, WGCNA (Langfelder and Horvath 2008) suggests using variance stabilizing transformation of RNA-Seq data before proceeding with the analysis.

7 Statistical Analysis

At this point of the pipeline, data appear in a matrix where each entry represents the expression level for a gene in one sample.

The normalized matrix can be used as input for the following steps of the analysis: differential expression, co-expression analysis or exploratory analysis like clustering and data visualization.

At this point, the normalized matrix can be treated in the same way as matrices originating from microarray technologies.

One difference has to be taken into consideration: values from RNA-Seq data are discrete measures because they are based on counts of the reads, while microarray data are continuous measures based on intensity values (Fang et al. 2012).

RNA-Seq data are characterized by two properties: the presence of extreme values and heteroscedasticity (relation between variance and mean of gene expressions).

For these reasons, data from RNA-Seq data are usually transformed in a logarithmic way or with other types of transformation like variance stabilizing transformation (Lin et al. 2008). Tools developed in a specific way for RNA sequencing do not need logarithmic transformation because they already take into account the typical distribution of the data counts.

8 Differential expression analysis

DE analysis allows to recognize genes whose expression is related to a trait of interest, such as those genes whose expression changes between conditions with enough statistical power. In this step, a statistical test is applied to each gene to determine whether we have enough statistical power to reject the null hypothesis that the gene is equally expressed in two or more conditions.

Differentially expressed genes provide information about the functions of genes under different conditions. From a systems biology perspective, the analysis of a set of DE genes can be integrated with information from different omics levels, leading to the identification of potential biological pathways involved in a process.

In RNA-Seq, this step is one of the most critical, for which a number of methods have been developed.

Each method is based on different assumptions regarding the distribution of the gene counts and on different statistical models. Some of them can deal with multifactorial analysis, others can be applied in experimental designs with no replicates, while still others allow for isoform detection and quantification (Mazzoni et al. 2015). Above all, the performances are dependent on the structure of the data.

Many tools have been tested with both real and simulated data sets. From these studies, the performances of the tools are strictly dependent on the properties of the dataset and on the experimental design (Zhang et al. 2014; Seyednasrollah et al. 2015).

The choice of the tool is fundamental. Taking into consideration that there is great variability in the maturity (Garber et al. 2011) of available computational tools, it is important that the user is aware of the main differences and makes a choice considering properties of the data like number of samples, replicates and heterogeneity of the dataset (Seyednasrollah et al. 2015).

Tools for differential expression can be classified in non-parametric tools that are not based on the assumption of the distribution of the gene counts, and the parametric tool where gene expression of the genes is assumed to have a specific distribution.

Among the non-parametric methods we find NOISeq and SAMSeq.

Both of them perform very well in terms of control of false positives, but they have opposite characteristics: NOISeq is too conservative with a high number of replicates, while SAMSeq needs more replicates for a good power of detection and its performances are strictly related to the data (Soneson and Delorenzi 2013; Seyednasrollah et al. 2015).

Among parametric methods, the best performing tools are DESeq, edgeR (Robinson et al. 2010) and BaySeq (Hardcastle and Kelly 2010), which appear to be similar in terms of accuracy, control of the number of false positives and sensitivity (Zhang et al. 2014; Kvam et al. 2012).

In datasets with small sample size, the best tools turned out to be Limma and DESeq.

DESeq proved to be the most conservative, while edgeR has a higher power of detection and Limma is the most robust with strong consistency of the results across heterogeneous datasets (Seyednasrollah et al. 2015; Soneson and Delorenzi 2013).

DESeq's successor, DESeq2, has a higher power of detection, but is less precise (Seyednasrollah et al. 2015).

BaySeq, based on Bayesian methodology, showed good performances in different cases but is strongly dependent on the dataset structure (Seyednasrollah et al. 2015; Soneson and Delorenzi 2013).

Finally, one of the most prominent tools, Cuffdiff2, has good performances but poor power of detection at the gene level (Seyednasrollah et al. 2015; Zhang et al. 2014).

However, one of the main advantages of Cuffdiff2 is the possibility to compute expression changes at the gene and transcript levels.

In the case of complex experimental designs, where more than one variable can be correlated to the gene expression levels, the possibility of accounting for those variables in the model is very important.

DESeq, DESeq2, edgeR, Limma and NOISeq allow for performing multifactorial analysis (Love et al. 2014; Robinson et al. 2010; Ritchie et al. 2015; Tarazona et al. 2012). Thanks to these tools, it is very easy to deal with very complex experimental designs, even for less experienced users.

Typically, the user gives as input the linear model that the tool will fit before computing the contrast. The basic model is:

$$y_{\mathrm{i}} = \mathrm{covariate}_1 + \mathrm{covariate}_2 + \mathrm{covariate}_n + \mathrm{trait\_of\_interest}$$

where y_{i} is the gene normalized gene counts for gene i across all the samples, covariate 1 to n represents potential confounding effects that have to considered during the test and the trait of interest is the covariate, which has to be performed for the differential expression analysis.

The program will fit many models as the number of genes given in input ($i = 1$ to t), where t is the number of genes to be tested.

For DESeq, edgeR and Limma, very extensive explanations of the tools are provided together with the manuals, making their use and the interpretation of the results even easier (Seyednasrollah et al. 2015).

	baseMean	log2FoldChange	lfcSE	stat	pvalue	padj
	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>	<numeric>
ENSBTAG00000000010	956660606	0.02814910	0.2112653	0.1332405	0.8940032	0.9976981
ENSBTAG00000000012	212395995	-0.12010838	0.2113485	-0.5682953	0.5698344	0.9976981
ENSBTAG00000000013	273204564	-0.23214247	0.2059338	-1.1272672	0.2596295	0.9851721
ENSBTAG00000000014	1495445436	0.16548246	0.1891041	0.8750867	0.3815267	0.9976981
ENSBTAG00000000015	70768749	0.04718367	0.2770936	0.1702806	0.8647894	0.9976981
...	...	...	...	...	...	...
ENSBTAG00000048293	53897117	-0.07879336	0.2846027	-0.2768539	0.7818923	0.9976981
ENSBTAG00000048296	74821939	-0.32501805	0.2840713	-1.1441424	0.2525646	0.9851721
ENSBTAG00000048306	21861174	-0.23933382	0.2682570	-0.8921811	0.3722959	0.9976981
ENSBTAG00000048308	166023977	-0.20215832	0.2682399	-0.7536474	0.4510610	0.9976981
ENSBTAG00000048314	187840008	-0.09321117	0.2674218	-0.3485549	0.7274235	0.9976981

Fig. 5 DESeq2 results from a differential expression analysis performed on bovine RNA-Seq data. *BaseMean*, mean of normalized counts for all samples; *log2FoldChange*, estimate of the gene expression change for the trait analysed (reported in a $\log^2$ scale); *lfcSE*, standard error associated to the estimate; *stat*, Walt test statistic; *p*-value, *p*-values obtained from the Walt test; *padj*, *p*-values adjusted for multiple testing (Benjamini–Hochberg procedure)

9 Interpretation of DE Analysis Results

The output file generated by most of the tools from a differential expression analysis consists of a list of genes or features followed by different parameters obtained from the statistic tests (Fig. 5).

The important parameters that are obtained from a differential expression analysis and that generally are presented in the final results file are the estimated fold change, the associated *p*-value and the *p*-value adjusted for multiple testing. The estimated fold change is the effect size estimate. The effect size estimate represents how much the expression of a gene changes due to the condition for which the contrast has been computed. Usually, this parameter is in a base 2 logarithmic scale. The tools compute also a statistic test that can be, for example, a Walt test, a likelihood ratio test or a Bayes statistic in order to obtain a *p*-value associated to the estimates.

Together with the *p*-value, the related adjusted *p*-value is also usually computed. The adjusted *p*-value is the statistic significance after multiple testing corrections. Usually the multiple testing is based on false discovery rate, but each tool gives the possibility to choose between different methods (Robinson et al. 2010; Anders and Huber 2010; Ritchie et al. 2015; Love et al. 2014).

The adjusted *p*-values give information about the significance of the gene expression change.

In general, to evaluate the differentially expressed genes, two thresholds should be set up; one for the adjusted *p*-value and another one for the fold change. In this way, it is possible to select genes whose change in expression is statistically significant and with a certain magnitude.

Conclusions

In this review, we have summarized all basic steps of the pipeline for RNA-Seq data analysis focusing on the steps that allow to check and get rid of the biases that can arise from RNA-Seq data.

In order to obtain accurate results, it is really important to remove potential sources of biases. The choice of the right tool, as well as the choice on how to identify problems in the data and to get rid of them, can have big impact on the final results.

This choice is not always easy and in order to perform a good analysis, it requires good knowledge about the tools available as well as about the RNA-Seq technology.

While for microarray analysis, the general standard to record and report microarray-based gene expression data has been defined in the MIAME guideline (Brazma et al. 2001), until now, no golden standard has been described for RNA-Seq data analysis.

One of the objectives of the FAANG project (http://www.faang.org/) is to establish a standard procedures for core assays, experimental protocols and also for RNA-Seq analysis pipeline in animal genomic research field.

In the absence of a commonly accepted standard procedure, the general overview presented in this review can help the reader in setting up the analytic pipeline. Furthermore, it can help to make the best choice in term of tools to use, thanks to the wide description of their characteristic and of the comparison of their performances.

In conclusion, this review can be useful for both educational purposes as well as for less experienced practitioners of animal genomic research who are dealing with RNA-Seq data.

Acknowledgments We thank Programme Commission on Health, Food and Welfare of the Danish Council for Strategic Research (Innovationsfonden) for financial support within the GIFT project.

References

Anders S, Huber W (2010) Differential expression analysis for sequence count data. Genome Biol 11(10):R106

Anders S, Pyl PT, Huber W (2014) HTSeq–A Python framework to work with high-throughput sequencing data. Bioinformatics btu638, 31(2):166–9

Andrews S (2010) FastQC: a quality control tool for high throughput sequence data., Reference Source

Benjamini Y, Speed TP (2012) Summarizing and correcting the GC content bias in high-throughput sequencing. Nucleic Acids Res gks001, 40(10):e72

Brazma A, Hingamp P, Quackenbush J, Sherlock G, Spellman P, Stoeckert C, Aach J, Ansorge W, Ball CA, Causton HC (2001) Minimum information about a microarray experiment (MIAME)—toward standards for microarray data. Nat Genet 29(4):365–371

Bullard JH, Purdom E, Hansen KD, Dudoit S (2010) Evaluation of statistical methods for normalization and differential expression in mRNA-Seq experiments. BMC Bioinform 11(1):94

Cochrane GR, Galperin MY (2010) The 2010 nucleic acids research database issue and online database collection: a community of data resources. Nucleic Acids Res 38(suppl 1):D1–D4

DeLuca DS, Levin JZ, Sivachenko A, Fennell T, Nazaire M-D, Williams C, Reich M, Winckler W, Getz G (2012) RNA-SeQC: RNA-seq metrics for quality control and process optimization. Bioinformatics 28(11):1530–1532

Dillies M-A, Rau A, Aubert J, Hennequet-Antier C, Jeanmougin M, Servant N, Keime C, Marot G, Castel D, Estelle J (2013) A comprehensive evaluation of normalization methods for Illumina high-throughput RNA sequencing data analysis. Brief Bioinform 14(6):671–683

Dobin A, Davis CA, Schlesinger F, Drenkow J, Zaleski C, Jha S, Batut P, Chaisson M, Gingeras TR (2013) STAR: ultrafast universal RNA-seq aligner. Bioinformatics 29(1):15–21

Engström PG, Steijger T, Sipos B, Grant GR, Kahles A, Rätsch G, Goldman N, Hubbard TJ, Harrow J, Guigó R (2013) Systematic evaluation of spliced alignment programs for RNA-seq data. Nat Methods 10(12):1185–1191

FAANG (Functional Annotation of Animal Genomes). http://www.faang.org/

Fang Z, Martin J, Wang Z (2012) Statistical methods for identifying differentially expressed genes in RNA-Seq experiments. Cell Biosci 2(1):26

Garber M, Grabherr MG, Guttman M, Trapnell C (2011) Computational methods for transcriptome annotation and quantification using RNA-seq. Nat Methods 8(6):469–477

García-Alcalde F, Okonechnikov K, Carbonell J, Cruz LM, Götz S, Tarazona S, Dopazo J, Meyer TF, Conesa A (2012) Qualimap: evaluating next-generation sequencing alignment data. Bioinformatics 28(20):2678–2679

Grabherr MG, Haas BJ, Yassour M, Levin JZ, Thompson DA, Amit I, Adiconis X, Fan L, Raychowdhury R, Zeng Q (2011) Full-length transcriptome assembly from RNA-Seq data without a reference genome. Nat Biotechnol 29(7):644–652

Hansen KD, Irizarry RA, Zhijin W (2012) Removing technical variability in RNA-seq data using conditional quantile normalization. Biostatistics 13(2):204–216

Hardcastle TJ, Kelly KA (2010) baySeq: empirical Bayesian methods for identifying differential expression in sequence count data. BMC Bioinformatics 11(1):422

Kim D, Salzberg SL (2011) TopHat-Fusion: an algorithm for discovery of novel fusion transcripts. Genome Biol 12(8):R72

Kroll KW, Mokaram NE, Pelletier AR, Frankhouser DE, Westphal MS, Stump PA, Stump CL, Bundschuh R, Blachly JS, Yan P (2014) Quality control for RNA-seq (QuaCRS): an integrated quality control pipeline. Cancer Inform 13(Suppl 3):7

Kvam VM, Liu P, Si Y (2012) A comparison of statistical methods for detecting differentially expressed genes from RNA-seq data. Am J Bot 99(2):248–256

Langfelder P, Horvath S (2008) WGCNA: an R package for weighted correlation network analysis. BMC Bioinform 9(1):559

Lassmann T, Hayashizaki Y, Daub CO (2011) SAMStat: monitoring biases in next generation sequencing data. Bioinformatics 27(1):130–131

Lin SM, Du P, Huber W, Kibbe WA (2008) Model-based variance-stabilizing transformation for Illumina microarray data. Nucleic Acids Res 36(2):e11–e11

Love MI, Huber W, Anders S (2014) Moderated estimation of fold change and dispersion for RNA-seq data with DESeq2. Genome Biol 15(12):1–21

Mazzoni G, Kogelman L, Suravajhala P, Kadarmideen H (2015) Systems genetics of complex diseases using RNA-sequencing methods. Int J Biosci Biochem Bioinform 5(4):264

Mortazavi A, Williams BA, McCue K, Schaeffer L, Wold B (2008) Mapping and quantifying mammalian transcriptomes by RNA-Seq. Nat Methods 5(7):621–628

Mutz K-O, Heilkenbrinker A, Lönne M, Walter J-G, Stahl F (2013) Transcriptome analysis using next-generation sequencing. Curr Opin Biotechnol 24(1):22–30

Oshlack A, Wakefield MJ (2009) Transcript length bias in RNA-seq data confounds systems biology. Biol Direct 4(1):14

Oshlack A, Robinson MD, Young MD (2010) From RNA-seq reads to differential expression results. Genome Biol 11(12):220

Risso D, Schwartz K, Sherlock G, Dudoit S (2011) GC-content normalization for RNA-Seq data. BMC Bioinform 12(1):480

Ritchie ME, Phipson B, Wu D, Hu Y, Law CW, Shi W, Smyth GK (2015) limma powers differential expression analyses for RNA-sequencing and microarray studies. Nucleic Acids Res gkv007, 43(7):e47

Robinson MD, Oshlack A (2010) A scaling normalization method for differential expression analysis of RNA-seq data. Genome Biol 11(3):R25
Robinson MD, McCarthy DJ, Smyth GK (2010) edgeR: a Bioconductor package for differential expression analysis of digital gene expression data. Bioinformatics 26(1):139–140
Seyednasrollah F, Laiho A, Elo LL (2015) Comparison of software packages for detecting differential expression in RNA-seq studies. Brief Bioinform 16(1):59–70
Soneson C, Delorenzi M (2013) A comparison of methods for differential expression analysis of RNA-seq data. BMC Bioinform 14(1):91
Tarazona S, García F, Ferrer A, Dopazo J, Conesa A (2012) NOIseq: a RNA-seq differential expression method robust for sequencing depth biases. EMBnet J 17(B):18–19
Trapnell C, Hendrickson DG, Sauvageau M, Goff L, Rinn JL, Pachter L (2013) Differential analysis of gene regulation at transcript resolution with RNA-seq. Nat Biotechnol 31(1):46–53
Wang Z, Gerstein M, Snyder M (2009) RNA-Seq: a revolutionary tool for transcriptomics. Nat Rev Genet 10(1):57–63
Wang L, Wang S, Li W (2012) RSeQC: quality control of RNA-seq experiments. Bioinformatics 28(16):2184–2185
Williams AG, Thomas S, Wyman SK, Holloway AK (2014) RNA¯seq data: challenges in and recommendations for experimental design and analysis. Curr Protoc Hum Genet 11.13. 11–11.13. 20
Williams CR, Baccarella A, Parrish JZ, Kim CC (2016) Trimming of sequence reads alters RNA-Seq gene expression estimates. BMC Bioinform 17(1):1
Wysoker A, Tibbetts K, Fennell T (2012) Picard. http://picard.sourceforge.net.
Zhang ZH, Jhaveri DJ, Marshall VM, Bauer DC, Edson J, Narayanan RK, Robinson GJ, Lundberg AE, Bartlett PF, Wray NR (2014) A comparative study of techniques for differential expression analysis on RNA-Seq data 9(8):e103207
Zhao S, Fung-Leung W-P, Bittner A, Ngo K, Liu X (2014) Comparison of RNA-Seq and microarray in transcriptome profiling of activated T cells. PLoS One 9(1)
Zheng W, Chung LM, Zhao H (2011) Bias detection and correction in RNA-Sequencing data. Bmc Bioinform 12(1):290